Predicting complex user behavior from CDR based social networks

Stephen Dipple, Casey Doyle, Zala Herga, Boleslaw K. Szymanski, Gyogry Korniss

Rensselaer

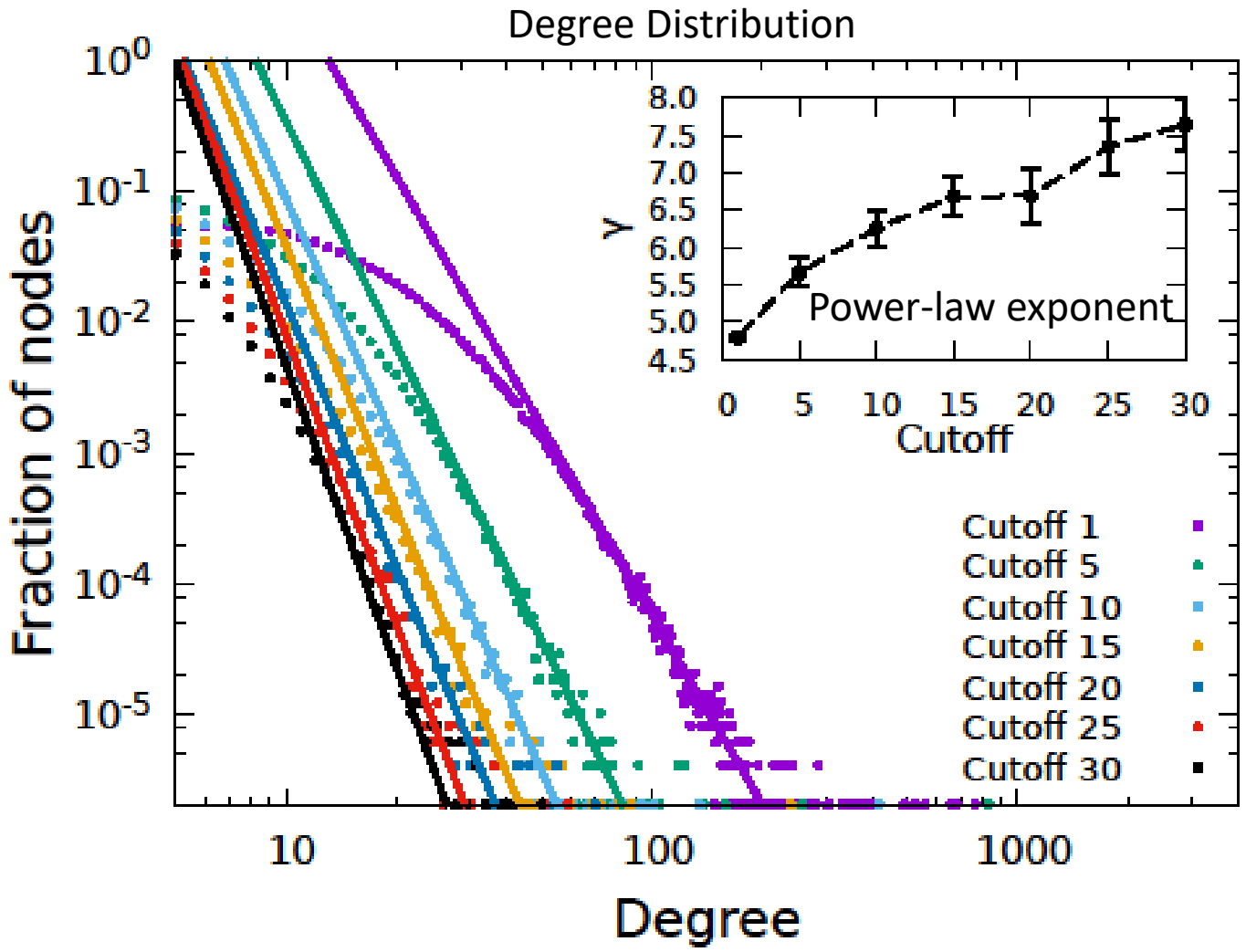
Data Collection

- We used data collected over a three month period, this includes all texts and calls made by any given person in the network, with whom the communication was made, and timestamp, location, and call duration.
- For privacy reasons, we cannot disclose absolute locations, or exact number of participants, just the order of magnitude, $N \sim 500{,}000$.
- In addition we collected the age, gender, and home district of each person along with whether during the three month period that person defaulted on paying their cell phone bill.

Building the Network

First we build an unweighted graph using a threshold or cutoff required to create a link based on number of communications.

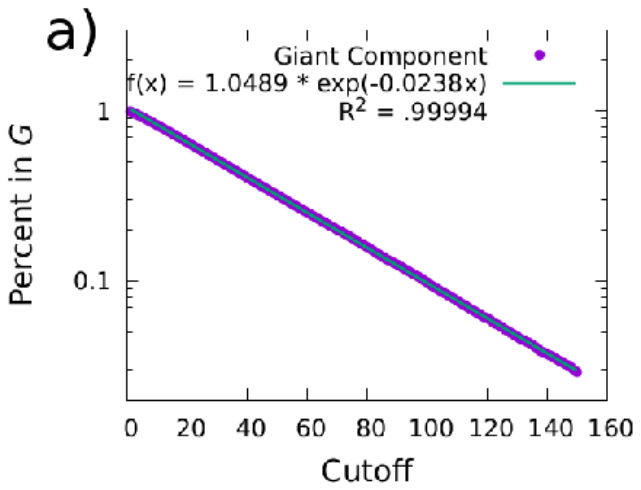
Various cutoffs produce similar degree distributions with small changes in the power-law exponent.

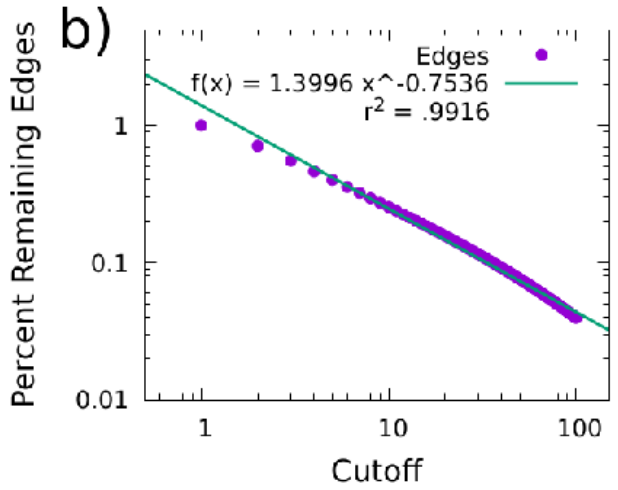


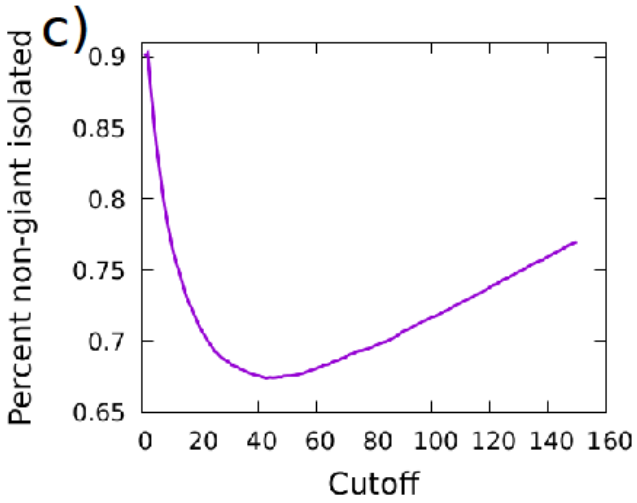
Testing Resilience

Here we examine the giant and secondary components of the network for various values of cutoffs.

This reveals that there are few critical links connecting components or that such links have a variety of number of communication so not all are removed at the same time by the small change of cutoff.







Shortest path

When calculating shortest path we wish to avoid using a cutoff, but still prevent a shortest path from using a low level communication link unless necessary. To do this we define the following distance between nodes.

$$d_{st} = \frac{w_{avg}}{w_{st}}$$

Here, w_{avg} is the average level of communication, represented by link's weight of all links, and w_{st} is the sum of weights of links on the path from the source node s to the target node t. Note that in general w_{ij} does not exclusively equal w_{ji} and a link from i to j may not have a link going from j to i.

Centrality Measures

Harmonic centrality gives a measure of distance from any node to another node.

$$C(i) = \sum_{i \neq j} \frac{1}{l_{ij}}$$

Here l_{ij} is the shortest distance between node i and j.

The average centrality of nodes is 4.61 while the standard deviation is 1.84.

Comparing against random graphs

A fair comparison of distances in order to better judge our centrality measures can be obtained by doing a rewiring of the network. Here we simply randomize the target of a link while preserving nodes' degrees. This process preserves out degree, but not in degree.

	Real	Random
Avg	4.61	4.11
Std	1.84	1.20

	Real	Random
Avg Shortest Path	.311	.300
Diameter	6.24	4.35

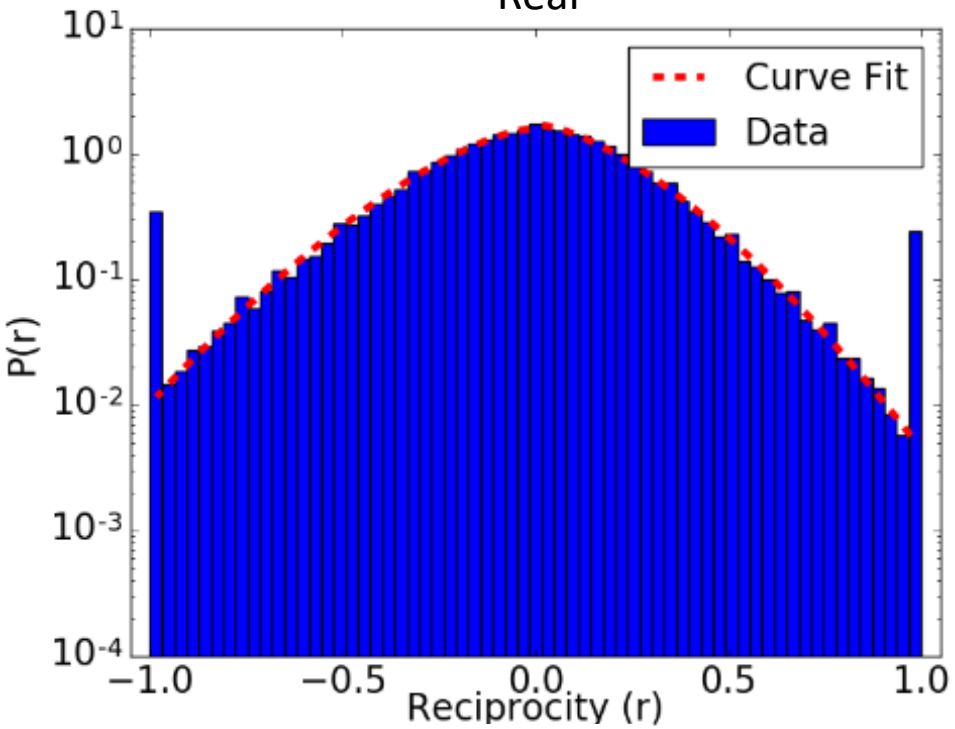
Here we can see nodes in a random graph are more central and have fewer extremes. Interestingly, the average shortest path is mostly unchanged, but the diameter changes significantly.

Reciprocity

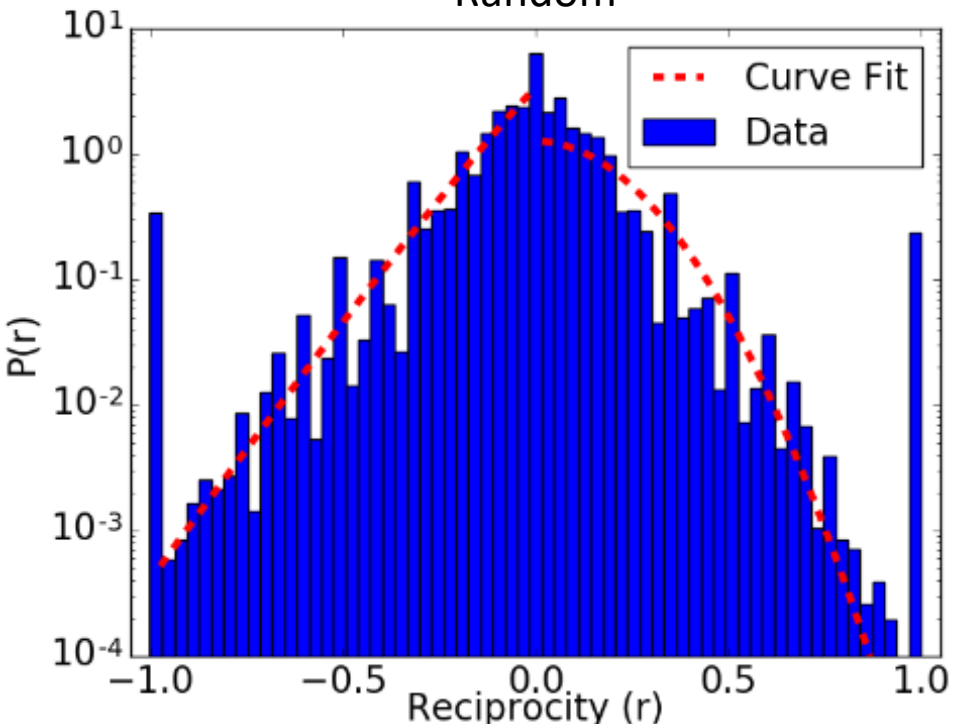
Here reciprocity describes to what extent communications are one way. We measure reciprocity using

$$R(i) = \frac{1}{k_i} \sum_{j \in N(i)} \frac{w_{ij} - w_{ji}}{w_{ij} + w_{ji}}$$

Real



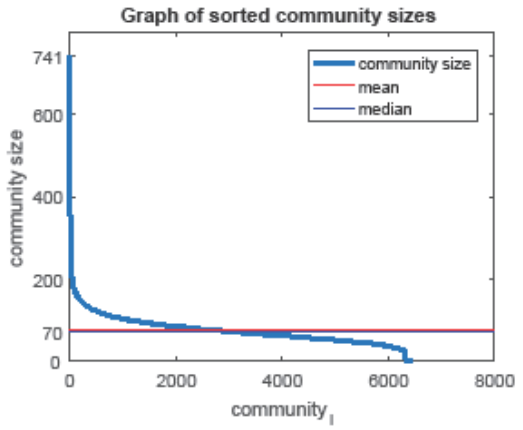
Random

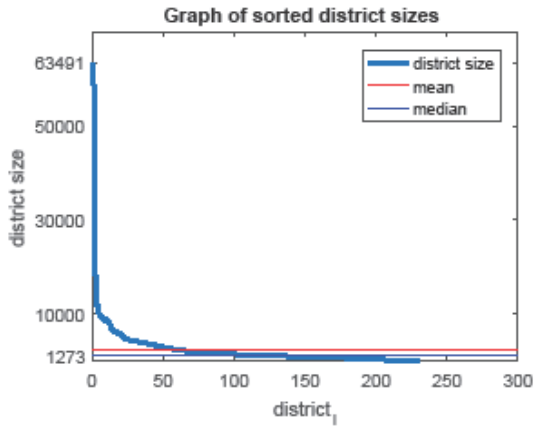


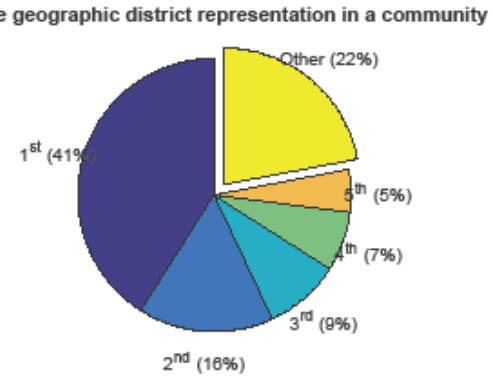
Communities

Here, we use GANXiS(SLPA) community detection algorithm to determine community structure in the network. This algorithm does well at recovering disjointed and overlapping communities.

The average community size is about 70 while average district size is about 2,380.

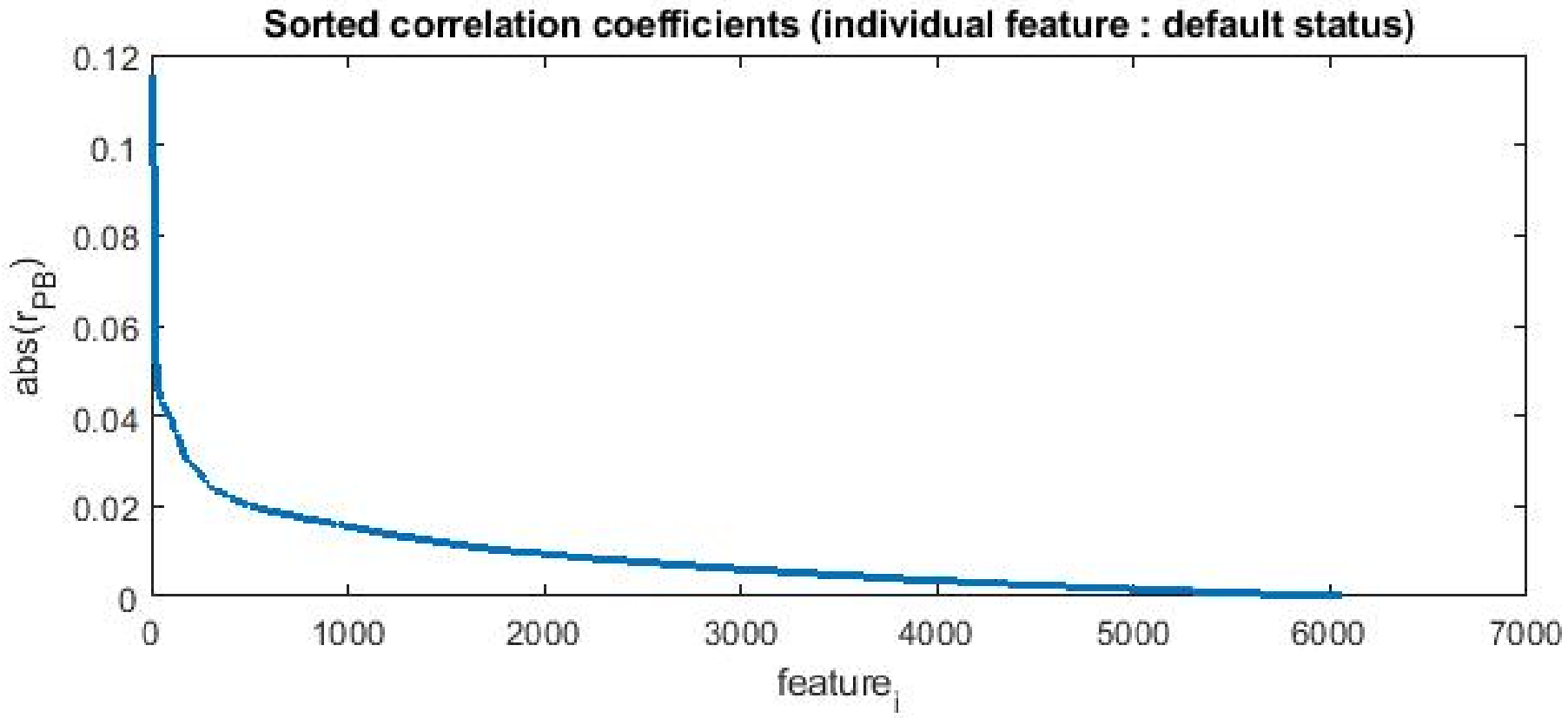






Characteristic Features

With about 6000 features, the average correlation to users defaulting on their cell phone bills is -0.006 with a max of 0.1155



Logistic Regression

To do our predictive modeling we use the following regression equation.

$$P(Y) = \frac{1}{1 + e^{-\beta_0 - \beta X}}$$

Here Y is our dependent variable vector of size *n*, X is a matrix with size *n* by *p* where the rows corresponds to observations and columns correspond to the predictor variables (our features), and β is our regression coefficients.

Logistic Regression

We can apply this model for various feature set size. We begin with 7 features, including 5 network measures and 2 location. All features, except for call duration which has a p value of 0.97, produces a p value ≤ 0.01.

Our most extensive model includes all 6048 features. Difficulties with such model include biased tradeoff between model simplicity vs accuracy. To balance this we create an additional model that break features into sets of linearly independent components.

Principal Component Analysis

Principal Component Analysis breaks these feature into set of linearly independent components. The best single component can account for 20% of the variability. The second only does 7%. The first 30 components account for 42% of variability. The first 500 components account for 66%.

We then produce two models, pca-30 which uses the best 30 components and pca-500. Lastly we produce a model pval-05 which removes all components from pca-500 that have a p-value over 0.5

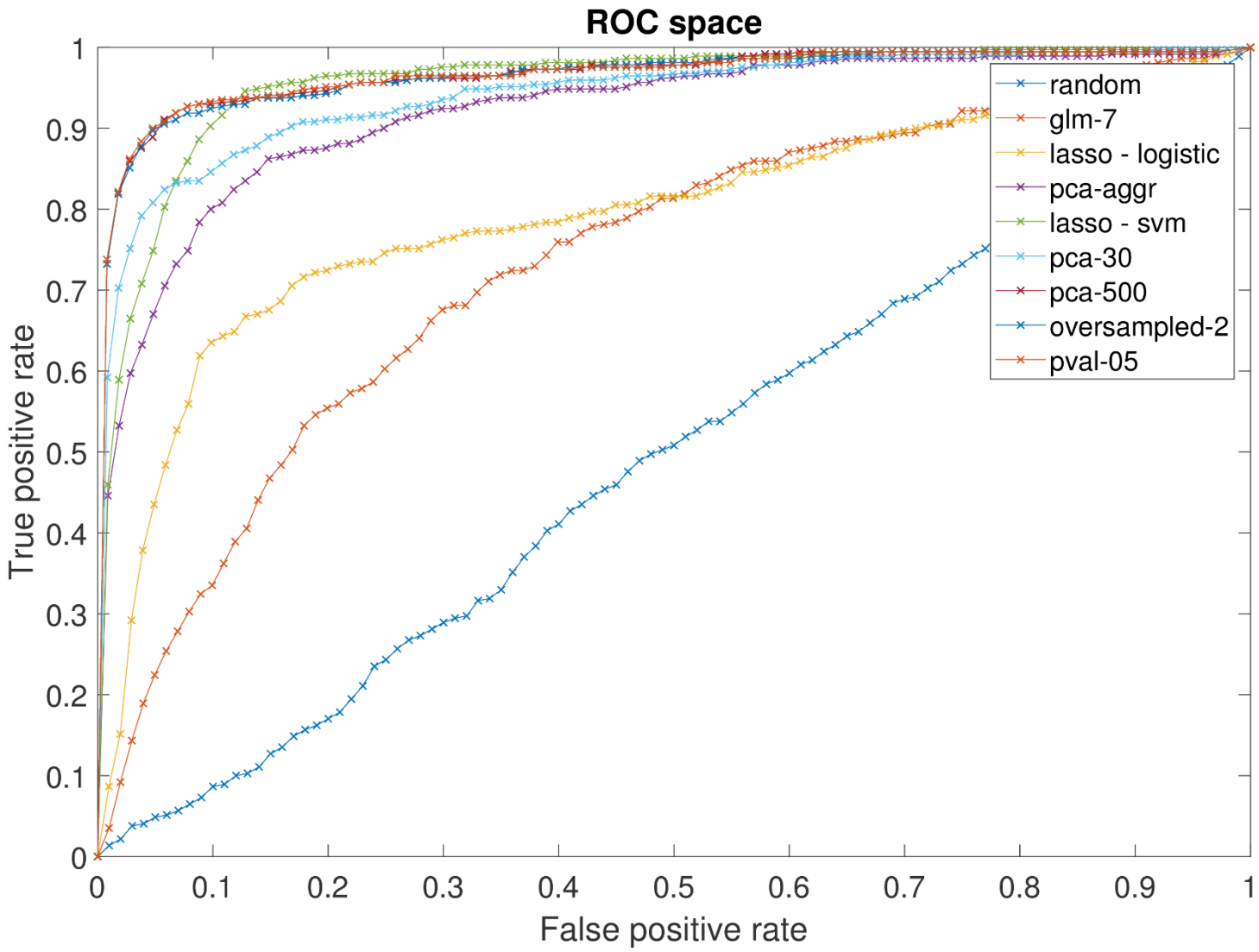
Feature Reduction

Defaulting users make up only 0.25% of the network which poses difficulty when performing data-mining techniques. One way to overcome this is by oversampling the data using a tunable multiplication factor. In this case we only keep the model with the best performing factor.

We also use Lasso-regression which employs a penalty term in the log-likelihood function to reduce coefficients of some features to zero based on a tunable parameter. Again we keep the model with the best performing factor.

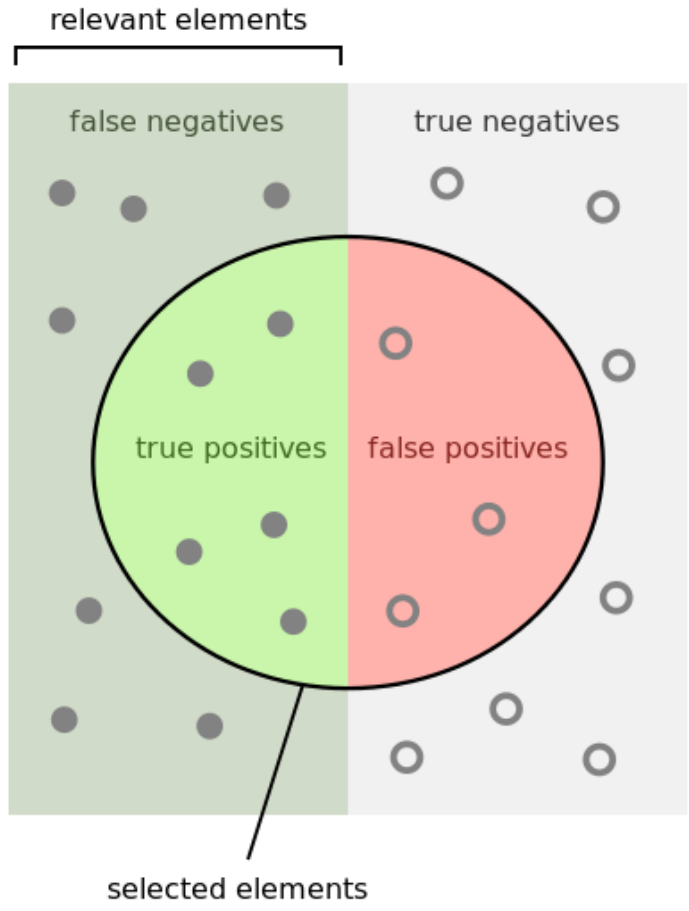
Evaluations

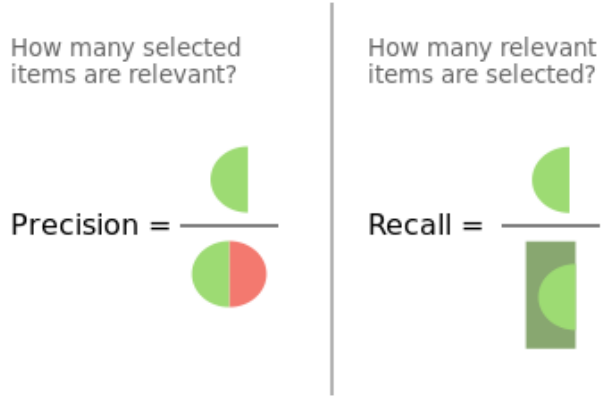
We train the selected predictive models on a random set of 70% of the network. We then compare the predictions generated by models trained on the remaining 30%. All models preform significantly better than random in blue. The best are pval-05 and pca-500.



Evaluations

Model	Recall	Fall-out	Precision
Random	0.060	0.0501	0.003
Glm-7	0.224	0.0495	0.012
Lasso-Logistic	0.484	0.0490	0.023
PCA-aggr	0.676	0.0484	0.036
Lasso-svm	0.749	0.0482	0.040
PCA-30	0.810	0.0480	0.043
PCA-500	0.889	0.0478	0.047
Oversampled-2	0.897	0.0478	0.047
Pval-05	0.900	0.0477	0.048





https://en.wikipedia.org/wiki/Precision_and_recall#/media/File:Precisionrecall.svg

Evaluations

Using PCA-500 we remove a single component to examine how the lack of that component effects recall and precision. The correspondent feature, which is the group of features focusing on unique frequent correspondents during various time frames.

Model	Recall	Precision	Δ Recall	Δ Precision
Full set	0.9	0.048		
Consumption	0.88	0.047	0.02	0.001
Correspondent	0.58	0.031	0.32	0.017
Reciprocated	0.88	0.046	0.02	0.002
Mobility	0.88	0.046	0.02	0.002
Network	0.88	0.05	0.02	-0.002
Cell Tower PD	0.89	0.050	0.01	-0.002
Only Correspondent	0.86	0.049		

Correspondent Features

Correspondent Features group contain 2543 features with examples being number of calls/text/messages sent on a day of the week, week, at work, or in the evening.

The best preforming features are the number of unique correspondents a person has on select days near winter holidays, 12/23-12/27.

Conclusion

The best model, pval-05 in this case has a recall of 0.9 with a fall-out of 0.0477.

The overwhelmingly strong features are the numbers of unique contacts with which the user interacted over a certain period of time, the best feature overall is the number of unique contacts interacted with over the winter holiday.

This suggests that the timing of communication has more importance than volume or duration. In addition this simple metric preforms than our more advanced metrics based on network characteristics.